Модуль управления ПО "Скала^р Геном"

Руководство администратора

Листов 15

2024

СОДЕРЖАНИЕ

1 УСТАНОВКА ПРОГРАММЫ

Для установки и работы ПО "Скала^р Геном" требуется операционная система Альт 8 СП релиз 9, Альт 8 СП релиз 10, Astra Linux Special Edition 1.7.3 (Орёл), RedOS 7.3.

1.1 Минимальные требования к устройству (виртуальной машине)

- **CPU**: от 4 ядер;
- **RAM**: от 16 Гб;
- **ROM**: от 100 Гб SSD;
- **NET**: от 1 Гбит/с Ethernet.

1.2 Состав дистрибутива

Дистрибутив содержит скрипт установки **`genome-installer.run`**.

1.3 Используемые сетевые порты

Порты, необходимы для нормальной работы:

ПОРТ	ПРОТОКОЛ	НАПРАВЛЕНИЕ	НАЗНАЧЕНИЕ
22	SSH	I/O	**SSH**-подключение к модулю управления Геном
5432	TCP	I/O	**PostgreSQL** БД модуля управления Геном
48800	TCP	I/O	**REST API** модуля управления Геном
50888	TCP	I/O	Web-интерфейс модуля управления Геном
15000-16000	HTTP	I/O	**IPMI proxies** для управления серверами

1.4 Ход развёртывания

1.4.1 Открыть терминал.

1.4.2 Проверить тип и версию установленной ОС:

```
[root@genome-release-111 ~]# cat /etc/os-release
NAME="ALT SPServer"
VERSION="8.4"
ID=altlinux
VERSION_ID=8.4
PRETTY_NAME="ALT 8 SP Server (cliff)"
ANSI_COLOR="1;33"
CPE_NAME="cpe:/o:alt:spserver:8.4"
HOME_URL="https://basealt.ru/"
BUG_REPORT_URL="https://bugs.altlinux.org/"
```

1.4.3 Проверить наличие свободного места для установки модуля управления ПО "Скала^р Геном":

```
[root@genome-release-111 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
udevfs          5,0M   64K  5,0M   2% /dev
runfs           7,9G  768K  7,9G   1% /run
/dev/sdb4        58G  2,7G   56G   5% /
tmpfs           7,9G     0  7,9G   0% /dev/shm
tmpfs           4,0M     0  4,0M   0% /sys/fs/cgroup
/dev/sdb2       2,0G  4,8M  1,8G   1% /boot/legacy
/dev/sdb1       256M  5,8M  250M   3% /boot/efi
tmpfs           1,6G  4,0K  1,6G   1% /run/user/0
```

1.4.4 Перейти в корневой каталог:

```
[root@genome-release-111 ~]# cd /root/
```

1.4.5 Загрузить файл дистрибутива модуля управления ПО "Скала^р Геном" из репозитория.

Дождаться окончания загрузки:

```
[root@GNM2-0 ~]# scp genome-installer.run 192.168.186.121:/root
genome-installer.run                                          100% 8641MB 111.0MB/s   01:17
```

1.4.6 Вывести список файлов в директории и проконтролировать наличие файла **`genome-installer.run`** и прав доступа к нему:

```
[root@genome-release-111 ~]# ls -al
total 7556820
drwx------  8 root root       4096 мар 15 11:37 .
drwxr-xr-x 24 root root        271 мар 15 11:13 ..
drwx------  3 root root         17 дек  4 20:30 .ansible
drwxr-xr-x  2 root root         32 дек  4 20:31 .ansible_async
-rw-r--r--  1 root root        206 мар 15 11:24 .bash_history
-rw-------  1 root root        217 июн 19  2018 .bash_logout
-rw-------  1 root root        168 июн 19  2018 .bash_profile
-rw-------  1 root root        521 июн 19  2018 .bashrc
drwxr-xr-x  3 root root         19 дек  4 18:54 .cache
-rw-r--r--  1 root root 7738118404 мар 15 09:43 genome-installer.run
-rw-------  1 root root        181 июн 19  2018 .i18n
drwx------  2 root root         86 дек  4 18:54 .install-log
-rw-------  1 root root         45 июн 19  2018 .rpmmacros
drwx------  2 root root         43 мар 15 11:15 .ssh
-rw-------  1 root root        215 июн 19  2018 .tcshrc
drwx------  2 root root         28 дек  4 20:32 tmp
-rw-------  1 root root       6850 мар 15 11:37 .viminfo
-rw-r--r--  1 root root        171 дек  4 20:31 .wget-hsts
-rw-------  1 root root         20 июн 19  2018 .zlogout
-rw-------  1 root root         22 июн 19  2018 .zprofile
-rw-------  1 root root        128 июн 19  2018 .zshenv
-rw-------  1 root root        175 июн 19  2018 .zshrc
```

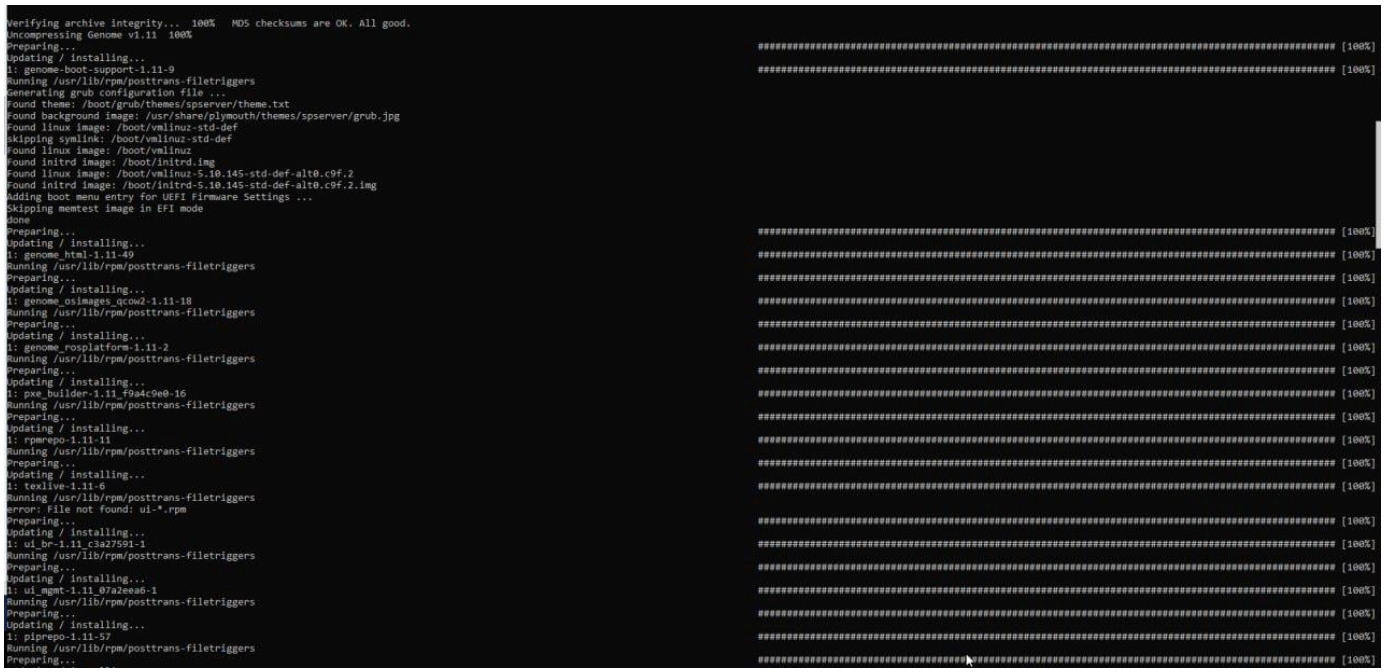
1.4.7 Разрешить выполнение файла **`genome-installer.run`**:

```
[root@cluster123-0 ~]# chmod +x genome-installer.run
```

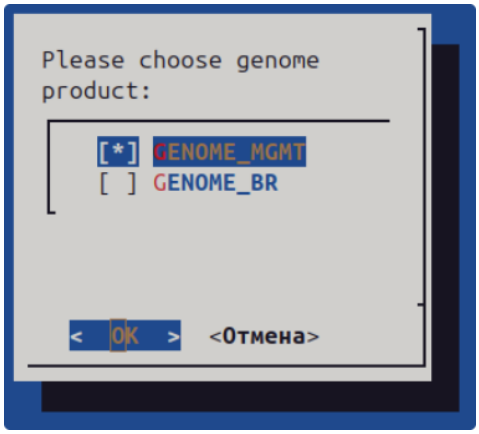
1.4.8 Запустить выполнение файла **`genome-installer.run`**:

```
[root@cluster123-0 ~]# ./genome-installer.run
```

Начнётся процесс распаковки и установки модулей:



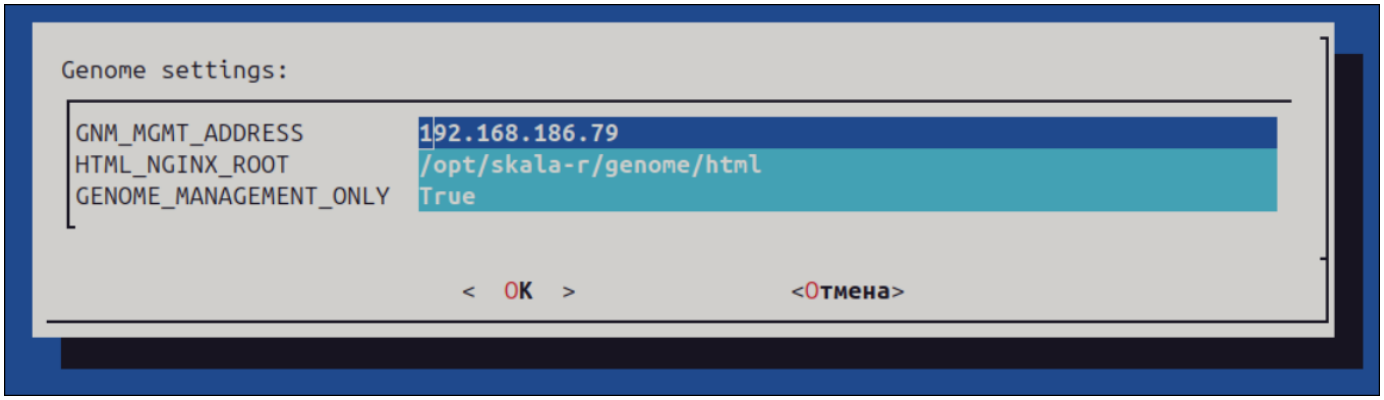
1.4.9 По завершении выполнения скрипта появится окно выбора ПО "Геном" для установки (здесь и далее, в случае необходимости, выбор подтверждается нажатием клавиши **ПРОБЕЛ** с последующим появлением символа «*» в соответствующем столбце):



GENOME_MGMT - установка модуля управления ПО "Скала^р Геном";

GENOME_BR - установка ПО "Скала^р Геном".

1.4.10 Выполнить ввод параметров развёртывания:



GNM_MGMT_ADDRESS – IP-адрес модуля управления ПО "Скала^р Геном";

HTML_NGINX_ROOT – путь до корневого каталога **html**, в котором находятся бинарные артефакты;

GENOME_MANAGEMENT_ONLY – признак установки только модуля управления ПО "Скала^р Геном".

1.4.11 После ввода всех необходимых данных нажать кнопку <**OK**>.

Должен начаться процесс настройки:

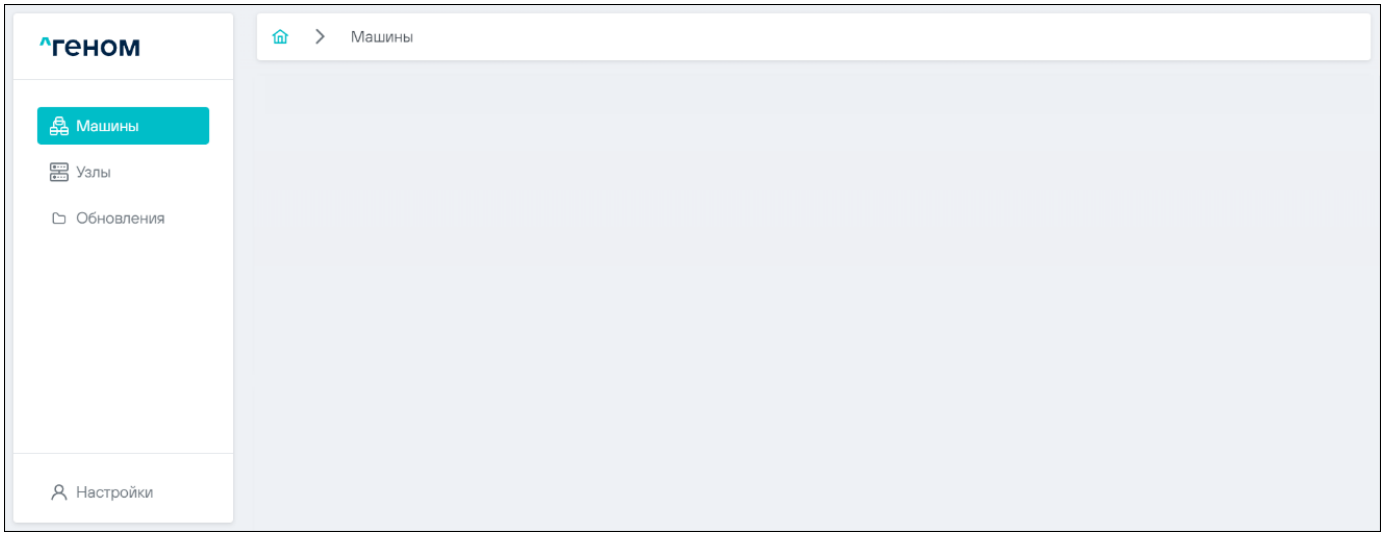
```
Do not install Genome BR product

PLAY [Installing Genome] ***********************************************************

TASK [Gathering Facts] *************************************************************
core/2.11/reference_appendices/interpreter_discovery.html for more information.
ok: [target]

TASK [Setting timezone] ************************************************************
changed: [target]

TASK [Recreate /root/.ssh if not exist] ********************************************
changed: [target]

TASK [Check if OpenSSH private key already exist] **********************************
ok: [target]

TASK [Check if OpenSSH public key already exist] ***********************************
ok: [target]

TASK [Get list of private keys] ****************************************************
ok: [target]

TASK [Write IdentityFile in SSH config] ********************************************

TASK [Generate an OpenSSH keypair with the default values (4096 bits, rsa)] ****
changed: [target]

TASK [Fix owner of the generated pub key] ******************************************
ok: [target]

TASK [Write public key] ************************************************************
changed: [target]

TASK [Postkey manipulations] *******************************************************
```

Дождаться завершения процесса настройки:

```
PLAY RECAP *********************************************************************
localhost                  : ok=10   changed=10   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
[root@alt84 ~]#
```

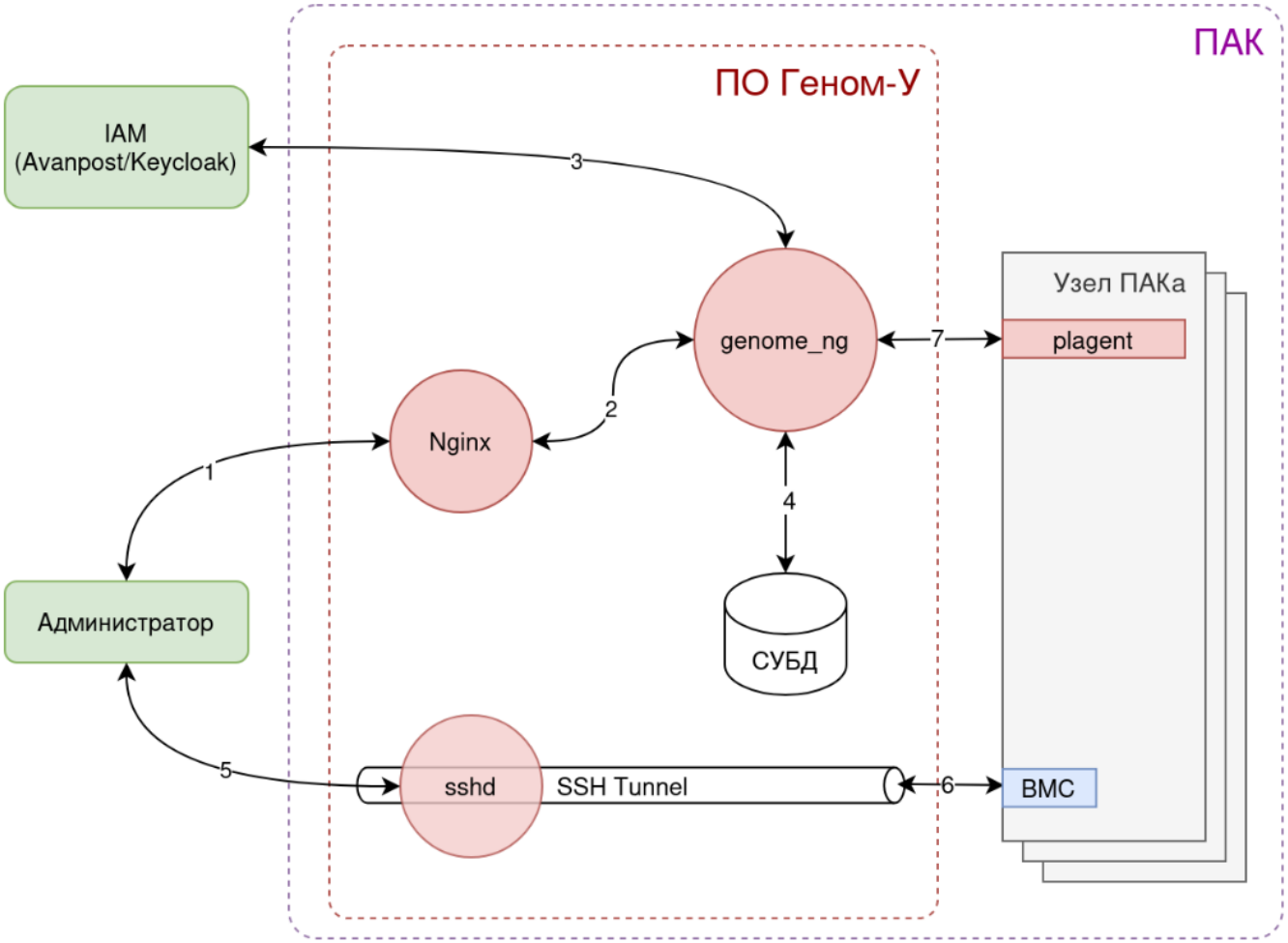
1.4.12 Перейти в браузер и ввести протокол https://IP-адрес, установленный в поле **GNM_MGMT_ADDRESS** п. 1.4.10 (порт 50888).

Должен открыться интерфейс модуля управления ПО "Скала^р Геном":



Модуль управления ПО "Скала^р Геном" считается успешно установленным, если после отображения интерфейса нет уведомлений об ошибках.

2 СХЕМА ВНУТРЕННИХ И ВНЕШНИХ ВЗАИМОДЕЙСТВИЙ



Назначение внутренних и внешних взаимодействий, используемые порты:

№ на схеме	Тип интерфейса	Назначение интерфейса
1	HTTPS TCP/52888	Пользовательский интерфейс Геном-У, аутентификация средствами **IAM**
2	HTTP TCP/48800	Передача управляющих команд из **UI**, без аутентификации
3	HTTP/HTTPS	Аутентификация и авторизация пользователей
4	TCP/5432	Взаимодействие с БД, аутентификация средствами СУБД
5	HTTPS TCP/15000-65535	Доступ к **Web UI** контроллера BMC узлов ПАКа через **SSH**-туннель, аутентификация на BMC
6	HTTPS TCP/443	
7	HTTPS TCP/7550	Взаимодействие с Агентом Генома (**plagent**), доступ по токену

3 КОНФИГУРАЦИЯ

Конфигурирование модуля управления ПО "Скала^р Геном" производится посредством редактирования файла **genome.json.**

Назначение полей файла конфигурации **genome.json** следующее:

Поле	Значение
mp_chroot	Путь для монтирования устройства, куда будет устанавливаться ОС на этапе **LiveCD**
genome_ip	Текущий IP-адрес Геном в сети Управления
chroot_env	Строка **chroot** с установкой необходимых переменных
genome_nginx_path	Путь до каталога **html**, в котором находятся бинарные артефакты
mbd_data_request_limit	Лимит запросов (для МБД.П)
genome_env	Раздел настройки установки Геном
GNM_PXE_ADDRESS	Текущий IP в сети **PXE**
GNM_MGMT_ADDRESS	Текущий IP в сети Управления
CHANGE_PG_SETTINGS	Изменение параметров **Postgres** по умолчанию (true / false)
PGHOST	IP-адрес подключения к **Postgres** Геном
PGUSER	Имя пользователя **Postgres**
PGDATABASE	Имя БД **Postgres**
PGPORT	Порт **Postgres**
PATH_PG	Путь к каталогу данных **Postgres**
PG_LISTEN_ADDR	Адрес сети, из которой разрешён доступ к **Postgres**
PG_TRUSTED_NETWORK	Доверенная подсеть **Postgres**, из которой не требуется ввод пароля
DNSMASQ_PXE_INTERFACE	Интерфейс, на котором запущены **DHCP** и **tftp** сервисы, предоставляемые **DNSMasq**
DNSMASQ_MGMT_INTERFACE	Интерфейс, на котором запущен **DNS** сервис, предоставляемый **DNSMasq**
USE_DNSMASQ	Включение и настройка **DNSMasq** при установке Геном (true / false)
DNSMASQ_FIRST_IP	Начальный IP-адрес **DHCP** пула адресов
DNSMASQ_LAST_IP	Конечный IP-адрес **DHCP** пула адресов
CREATE_PXE	Создание образа и настройка **PXE** (true / false)
HTML_NGINX_ROOT	Путь до каталога **html**, в котором находятся бинарные артефакты
PXE_TFTP_ROOT	Путь до каталога **tftp**
GENOME_MANAGEMENT_ONLY	Признак установки только модуля управления ПО "Скала^р Геном"
auth	Наименование блока, отвечающего за авторизацию
enabled	Включение авторизации в модуле управления ПО "Скала^р Геном" (true / false)

client_id	Идентификатор клиента из системы управления доступом
auth_server	Ссылка на систему управления доступом
realm	Область, которая включает в себя учётные записи пользователей, роли, группы и настройки авторизации
cert	Использование сертификата (true / false)

4 ЖУРНАЛИРОВАНИЕ

4.1 Виды журналов

В текущей версии реализованы два вида журналов: **journal** (Лог) и **audit** (Аудит). На данный момент все записи заносятся в оба журнала.

4.1.1 Лог

Записи журнала располагаются в директории

/opt/skala-r/var/log/genome_mgmt/journal.

Формат журнала – произвольный.

Пример записи:

```
2024-09-13T18:37:02+0300 user-genome-test INFO Приложение запущено
```

4.1.2 Аудит

Записи журнала располагаются в директории

/opt/skala-r/var/log/genome_mgmt/audit.

Аудит представлен в форматах **CEF** и **JSON**.

4.1.2.1 Формат CEF

Записи в формате **CEF** располагаются в директории

/opt/skala-r/var/log/genome_mgmt/audit/cef.

Пример записи:

```
CEF:0|Skala-r|genome_mgmt|1.13|10000040|Запуск: версия Геном.У|3|externalId=2
msg=Приложение запущено deviceProcessName=genome_core outcome=success
start=1726241821961 dhost=user-genome-test
```

Формат **CEF** имеет обязательные и необязательные поля.

Подстрока с обязательными полями:

CEF:0|Skala-r|genome_mgmt|1.13|20000040|Запуск: версия Геном.У|3|, где

CEF:0	Версия формата **CEF**
Skala-r	Название производителя
genome_mgmt	Название продукта

1.13	Версия продукта
20000040	Уникальный идентификатор категории события. **ID event** в таблице ресурсов
Запуск: версия Геном.У	Имя события
3	Важность события

Подстрока с расширенными полями:

externalId=2 msg=Приложение запущено deviceProcessName=genome_core outcome=success start=1726241821961 dhost=user-genome-test

Из записей в формате **CEF** на данный момент выводятся следующие параметры:

externalId	Идентификатор события
msg	Подробная информация о событии
deviceProcessName	Имя процесса, ассоциированного с событием
outcome	Результат события
start	Время возникновения события
dhost	**FQDN** или имя хоста получателя

4.1.2.2 Формат JSON

Записи в формате **JSON** располагаются в директории **/opt/skala-r/var/log/genome_mgmt/audit/json**.

Пример записи:

{"createdAt": "1726241825430", "userNode": "user-genome-test", "metamodelVersion": "1", "module": "genome_mgmt", "name": "Авторизация: Неизвестно (доступ запрещен)", "params": [{"name": "message", "value": "Ошибка доступа: 401 url /api/datasheet"}, {"name": "serviceVersion", "value": "1.13"}]}

Требования к формату не указаны.

На данный момент из записей в формате **json** выводятся следующие параметры:

createdAt	Время записи в журнал
userNode	Имя узла
metamodelVersion	Версия модели, на данный момент выставлена в "1"
module	Название приложения
name	Строка с событием
params	Структуры с версией приложения и текстовым сообщением

4.2 Конфигурационный файл

Управление записями журналов производится с помощью конфигурационного файла **logging_mgmt_conf.yml**, расположенного в директории **/opt/skala-r/genome/python-modules/lib/python3/site-packages/genome_ng/logging_mgmt_conf.yml**.

Пример:

```
# Все ключи являются необязательными

version: 1

# Аудит
audit:
  # Уровень сообщений, которые логируются DEBUG|INFO|WARNING|ERROR|CRITICAL
  level: INFO

  # Параметры форматов записи логов
  formatters:

    # Формат CEF
    cef:
      # Вкл/выкл запись логов в формате CEF
      enable: true

      # Управление ротированием файлов логов
      rotate:
        filepath: "/opt/skala-
r/var/log/genome_mgmt/audit/cef/genome_mgmt_audit.log"
        maxBytes: 104857600  # максимальный размер файла логов
        backupCount: 5       # количество сохраненных логов "<filepath>.<n>"
по maxBytes

    # Формат JSON (ГосТех)
    json:
      enable: true
      rotate:
        filepath: "/opt/skala-
r/var/log/genome_mgmt/audit/json/genome_mgmt_audit.log"
        maxBytes: 104857600
        backupCount: 5

  # Фильтрация записей по группам/действиям/статусам событий
  # Например, ["11-15", "2"] – исключить с 11 по 15 включительно и 2 ID
  # Перечень срезов genome-core/docs/events
  filters:
    groups: []
    actions: []
    status: []

# Журналирование, аналогично параметрам аудита
journal:
  level: INFO
  formatters:
    text:
      enable: true
      rotate:
        filepath: "/opt/skala-
r/var/log/genome_mgmt/journal/genome_mgmt_journal.log"
        maxBytes: 104857600
        backupCount: 5
  filters:
```

```
    groups: []
    actions: []
    status: []
```

Для включения / выключения какого-либо вида журналов используется параметр **enable**.

Имена файлов с записями, а также путь до этих файлов, задаются параметром **filepath**.

Размер файлов и их количество можно указать параметрами **maxBytes** и **backupCount**.

Параметром **filters** производится фильтрация записей в журналах.

Для применения новых настроек после изменения конфигурационного файла необходимо перезапустить приложение.

4.3 Фильтрация записей

Как было указано, фильтрация записей в журналы осуществляется при помощи параметра **filters** в конфигурационном файле.

Фильтровать записи возможно по трём параметрам: **groups**, **actions** и **status**.

В коде находится скрипт, генерирующий документацию с описанием этих параметров.

4.3.1 Фильтрация по группам

Все потенциальные записи разделены на группы в зависимости от типа. У каждой группы есть имя, описание и идентификатор, по которому можно производить фильтрацию.

Описание параметра **groups**:

Атрибут GroupEnum	Имя группы	Текст подгруппы	ID подгруппы
unknown	Неизвестно	Неизвестно	0
app	Приложение	Приложение **genome_mgmt**	10
auth_login	Авторизация	Вход пользователя в систему	21
auth_logout	Авторизация	Выход пользователя из системы	22
updates_mgmt	Обновления	Добавление и удаление обновлений в Геном.У	31
updates_apply	Обновления	Применение обновлений	32
updates_info	Обновления	Получение информации об обновлениях	33
updates_internal	Обновления	Внутренняя логика работы с обновлениями	34
passport_generate	Паспорта	Генерация паспортов	41
passport_internal	Паспорта	Внутренняя логика работы	42
node_replace_hw	Замена узла	Регистрация **hardware**	51
node_replace_sw	Замена узла	Регистрация **software**	52
node_replace_bm	Замена узла	Регистрация **benchmark**	53
ipmi_create	IPMI	Создание **IPMI**	61
ipmi_check	IPMI	Проверка доступа **IPMI**	62
ipmi_info	IPMI	Получение IPMI данных	63

Атрибут GroupEnum	Имя группы	Текст подгруппы	ID подгруппы
node_info	Узлы	Получение информации об узлах	71
node_update	Узлы	Изменение статуса и серийного номера узла	72
pak_mgmt	ПАК	Регистрация и удаление	81
pak_info	ПАК	Получение информации об ПАК	82
cluster_decommissioning	Кластер	Вывод из эксплуатации	91
cluster_info	Кластер	Получение информации о кластере	92
cluster_service	Кластер	Работа с сервисами	93
cluster_pgbouncer	Кластер	Работа с **pgbouncer**	94
cluster_uuid	Кластер	Работа с **uuid**	95
version	Версия Геном.У	Версия Геном.У	200

4.3.2 Фильтрация по действиям

Каждой записи соответствует действие, которое зависит от того, что нужно сделать с объектом. У действий также есть имя и идентификатор, по которому можно фильтровать записи.

Описание параметра **actions**:

Атрибут ActionEnum	Имя действия	ID действия
read	Чтение	1
update	Редактирование	2
auth	Авторизация	3
start	Запуск	4
stop	Остановка	5
log_rotate	Ротация логов	6

4.3.3 Фильтрация по статусам

Каждой записи соответствует статус, обозначающий результат выполнения действия над объектом. У статусов также есть имя и идентификатор, по которому можно фильтровать записи.

Описание параметра **status**:

Атрибут StatusEnum	Текст статуса	ID статуса
success	Успешно	0
failed	Неуспешно	1
forbidden	Доступ запрещен	2

Таким образом, если необходимо, например, исключить все неуспешно завершённые события, в параметр **status** нужно добавить "1" и перезапустить приложение.

Если необходимо исключить все записи, связанные с заменой узла, в параметр **groups** нужно добавить "51-53" и перезапустить приложение.

4.4 Передача файлов журнала во внешние системы

Передача журналов во внешние системы осуществляется с помощью ПО **rsyslog**. Перед настройкой необходимо установить пакет **rsyslog**.

Типовой набор конфигурационных файлов для настройки **rsyslog** находится в директории **`/opt/skala-r/genome/python-modules/lib/python/site-packages/genome_ng/settings/rsyslog`**.

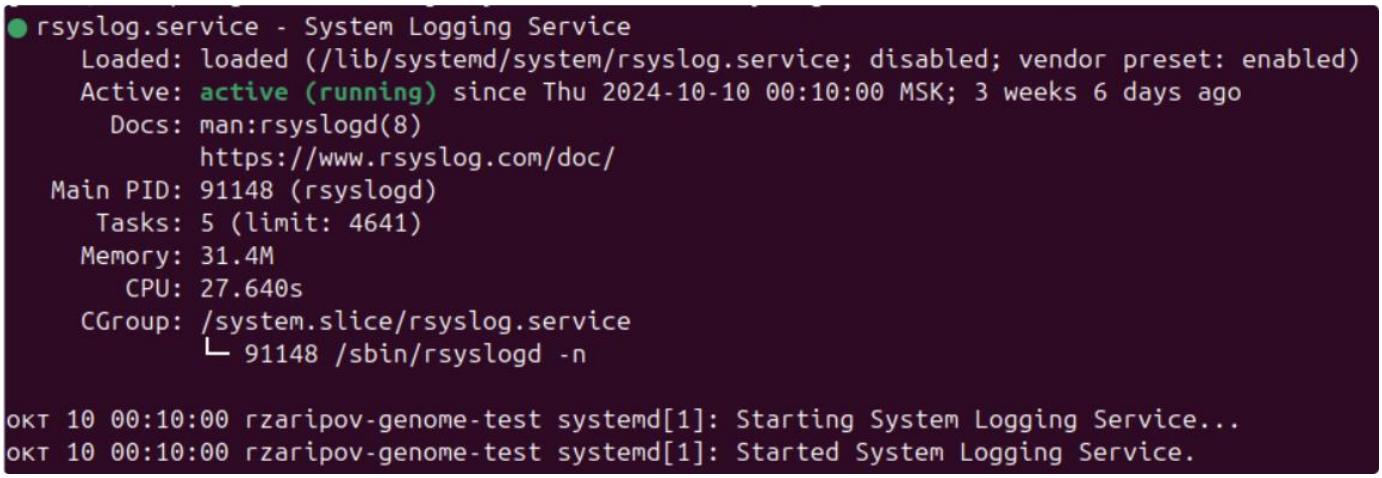
Для настройки необходимо скопировать указанный набор файлов в директорию **`/etc/rsyslog.d`** и указать адрес сервера во внешней **SIEM**-системе (**АС СВОИ**) в параметре **`Target`** в файла **`10_genome.conf`**.

После внесения изменений в конфигурационный файл необходимо запустить сервис **rsyslog** следующей командой:

```
systemctl start rsyslog
```

Убедиться, что сервис успешно запущен, можно с помощью команды:

```
systemctl status rsyslog
```



Статус должен быть **`active (running)`**.

5 ПРОВЕРКА СТАТУСА SSH ТУННЕЛЯ

Проверка производится с помощью утилиты

```
systemctl status tunnel_<ip>.service
```

где **`ip`** - IP-адрес BMC узлов ПАК из инсталляционной карты.

Статус должен быть **`active (running)`**.